

CLAIMS

We claim:

- 1 Claim 1 (original): An article of manufacture for use in a computer system for translating a path
2 expression in an object oriented query to a relational database outer join, said path expression
3 comprising a navigation path through a relationship in a schema, said article of manufacture
4 comprising a computer-useable storage medium having a computer program embodied in said
5 medium which causes the computer system to execute the method steps comprising:
 - 6 analyzing each path expression defined in each level of the object oriented query;
 - 7 identifying each path expression which can be a candidate for a translation to an outer
8 join;
 - 9 ordering the path expression starting with path expression defined in a FROM clause,
10 adding to the FROM clause path expression, each path expression identified as a candidate
11 for a translation to an outer join, and making the ordered path expressions as input to a select
12 operator for each level of the object oriented query;
 - 13 grouping the ordered path expressions sequentially based upon a source-target
14 dependency between ordered path expressions and based upon the identifications as a
15 candidate for a translation to an outer join;
 - 16 creating a quantifier for each path expression, said quantifier comprising a variable
17 representing a table in a relational database;
 - 18 replacing each grouped path expression with a corresponding quantifier and related
19 table in a relational database; and
 - 20 completing a translation of the object oriented query to a relational query.

1 Claim 2 (original): The article of manufacture of claim 1 wherein the embodied computer program
2 embodied in said medium can further cause the computer system to execute the method steps
3 comprising:

4 performing optimization on the grouped quantifiers, said optimization identifying
5 quantifiers which can be a candidate for a translation to an inner join;

6 generating an outer join for each quantifier which remains after optimization a
7 candidate for a translation to an outer join; and

8 generating an inner join for each quantifier which remains after optimization a
9 candidate for a translation to an inner join .

1 Claim 3 (original): The article of manufacture of claim 2 wherein the optimization identifies a
2 quantifier as a candidate for a translation to an inner join if a corresponding path expression is used
3 in a FROM clause.

1 Claim 4 (original): The article of manufacture of claim 2 wherein the optimization identifies a
2 quantifier as a candidate for a translation to an inner join if a LIKE, IN, or BETWEEN operator
3 exists in a WHERE clause containing a corresponding path expression.

1 Claim 5 (original): The article of manufacture of claim 2 wherein the optimization identifies a
2 quantifier as a candidate for a translation to an inner join if an EQUAL, LESS THAN, GREATER
3 THAN, LESS THAN OR EQUAL, GREATER THAN OR EQUAL, NOT EQUAL, or NOT NULL
4 operator exists in a WHERE clause.

1 Claim 6 (original): A method of translating a path expression in an object oriented query to a
2 relational database outer join, said path expression comprising a navigation path through a
3 relationship in a schema, said method comprising the steps of:

4 analyzing each path expression defined in each level of the object oriented query;

5 identifying each path expression which can be a candidate for a translation to an outer
6 join;

7 ordering the path expressions starting with path expressions defined in a FROM
8 clause, adding to the FROM clause path expressions, each path expression identified as a
9 candidate for a translation to an outer join, and making the ordered path expressions as input
10 to a select operator for each level of the object oriented query;

11 grouping the ordered path expressions sequentially based upon on a source-target
12 dependency between ordered path expressions and based upon the identifications as a
13 candidate for a translation to an outer join;

14 creating a quantifier for each path expression, said quantifier comprising a variable
15 representing a table in a relational database;

16 replacing each grouped path expression with a corresponding quantifier and related
17 table in a relational database; and

18 completing a translation of the object oriented query to a relational query.

1 Claim 7 (original): The method of claim 6 further comprising the steps of:

2 performing optimization on the grouped quantifiers, said optimization identifying
3 quantifiers which can be a candidate for a translation to an inner join;

4 generating an outer join for each quantifier which remains after optimization a
5 candidate for a translation to an outer join; and

6 generating an inner join for each quantifier which remains after optimization a
7 candidate for a translation to an inner join.

1 Claim 8 (original): The method of claim 7 wherein the optimization identifies a quantifier as a
2 candidate for a translation to an inner join if a corresponding path expression is used in a FROM
3 clause.

1 Claim 9 (original): The method of claim 7 wherein the optimization identifies a quantifier as a
2 candidate for a translation to an inner join if a LIKE, IN, or BETWEEN operator exists in a
3 WHERE clause containing a corresponding path expression.

1 Claim 10 (original): The method of claim 7 wherein the optimization identifies a quantifier as a
2 candidate for a translation to an inner join if an EQUAL, LESS THAN, GREATER THAN, LESS
3 THAN OR EQUAL, GREATER THAN OR EQUAL, NOT EQUAL, or NOT NULL operator exits
4 in a WHERE clause.

1 Claim 11 (original): A computer system for translating a path expression in an object oriented
2 query to a relational database outer join, said path expression comprising a navigation path through a
3 relationship in a schema, said computer system comprising:

4 computer program instructions for analyzing each path expression defined in each
5 level of the object oriented query;

6 computer program instructions for identifying each path expression which can be a
7 candidate for a translation to an outer join;

8 computer program instructions for ordering the path expressions starting with path
9 expressions defined in a FROM clause, adding to the FROM clause path expressions, each
10 path expression identified as a candidate for a translation to an outer join, and making the
11 ordered path expressions as input to a select operator for each level of the object oriented
12 query;

13 computer program instructions for grouping the ordered path expressions sequentially
14 based upon on a source-target dependency between ordered path expressions and based upon
15 the identifications as a candidate for a translation to an outer join;

16 computer program instructions for creating a quantifier for each path expression, said
17 quantifier comprising a variable representing a table in a relational database;

18 computer program instructions for replacing each grouped path expression with a
19 corresponding quantifier and related table in a relational database; and

20 computer program instructions for completing a translation of the object oriented
21 query to a relational query.

1 Claim 12 (original): The computer system of claim 11 further comprising:

2 computer program instructions for performing optimization on the grouped
3 quantifiers, said optimization identifying quantifiers which can be a candidate for a
4 translation to an inner join;

5 computer program instructions for generating an outer join for each quantifier which
6 remains after optimization a candidate for a translation to an outer join; and

7 computer program instructions for generating an inner join for each quantifier which
8 remains after optimization a candidate for a translation to an inner join.

1 Claim 13 (original): The computer system of claim 12 wherein the optimization identifies a
2 quantifier as a candidate for a translation to an inner join if a corresponding path expression is used
3 in a FROM clause.

1 Claim 14 (original): The computer system of claim 12 wherein the optimization identifies a
2 quantifier as a candidate for a translation to an inner join if a LIKE, IN, or BETWEEN operator
3 exists in a WHERE clause containing a corresponding path expression.

1 Claim 15 (original): The computer system of claim 12 wherein the optimization identifies a
2 quantifier as a candidate for a translation to an inner join if an EQUAL, LESS THAN, GREATER
3 THAN, LESS THAN OR EQUAL, GREATER THAN OR EQUAL, NOT EQUAL, or NOT NULL
4 operator exists in a WHERE clause.